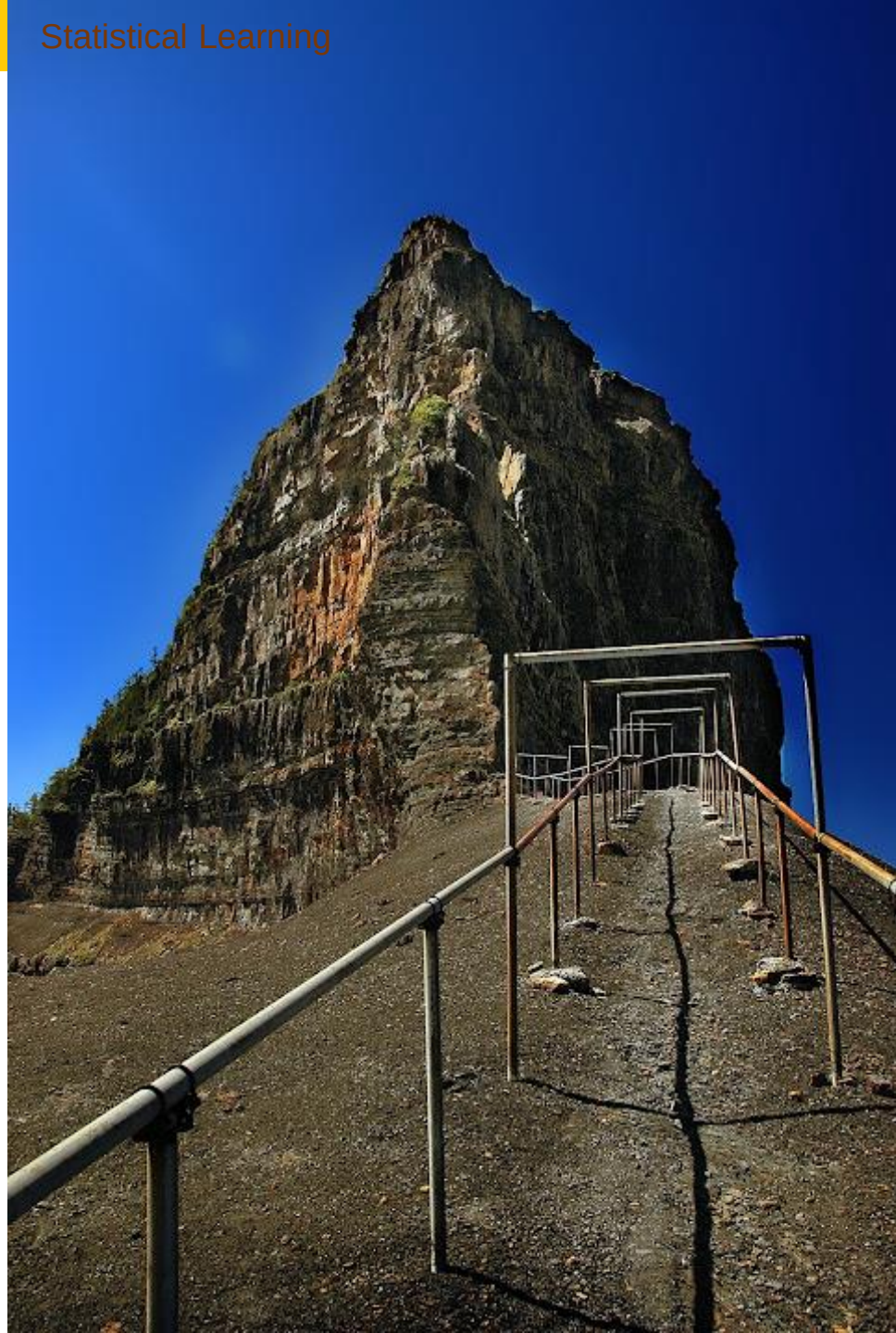
LINEAR MODELS FOR CLASSIFICATION (PART 3)

Hsin-Min Lu
盧信銘
台大資管系

Topics

- Introduction
- Discriminant Function
- Probabilistic Discriminative Models

- Laplace Approximation
- Bayesian Logistic Regression



THE LAPLACE APPROXIMATION

Topics

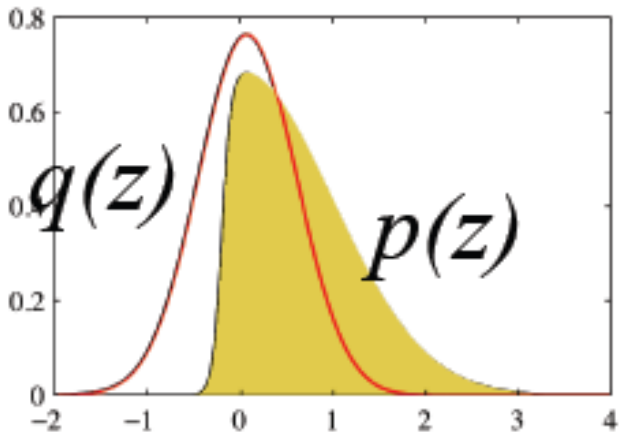
- Bayesian methods
- One-dimensional case
  - Example
- Multivariate case
- Weakness of Laplace approximation
- Model comparison and BIC

Bayesian methods

- Bayesian version of logistic regression is more complex than for linear regression
  - since posterior distribution over *w* is not Gaussian
- Need to introduce methods of approximation
- Approaches
  - Analytical Approximations
    - Laplace Approximation
  - Numerical Sampling (will not discuss here)

Laplace: One-dimensional case

- Consider single variable *z* with distribution *p(z)* defined by
  $p(z) = \frac{1}{Z} f(z)$,
  where $Z = \int f(z)dz$ is a normalization coefficient
  - *f(z)* could be a scaled version of *p(z)*
  - *p(z)* will be a pdf due to normalization
- Suppose that value of *Z* is unknown
- Goal is to find Gaussian approximation *q(z)* centered on the mode of the distribution *p(z)*



Taylor Expansion centered at Mode

- (Consider one-dimensional case) Finding the mode of *p(z)*
  - A point z_0 such that $p'(z_0) = 0$
  - Equivalently $\frac{df(z)}{dz}|_{z=z_0} = 0$
- Logarithm of Gaussian is a quadratic.
- So use Taylor expansion of $\ln f(z)$ centered at mode z_0
- $\ln f(z) \approx \ln f(z_0) - \frac{1}{2}A(z - z_0)^2$
- $A = -\frac{d^2}{dz^2}\ln f(z)|_{z=z_0}$
- First order term does not appear since z_0 is a local maximum

Final form of Laplacian (One Dimension)

- Approximation of *f(z)*, $\ln f(z) \approx \ln f(z_0) - \frac{1}{2}A(z - z_0)^2$
- Taking exponential $f(z) \approx f(z_0)\exp\left\{-\frac{A}{2}(z - z_0)^2\right\}$
- Normalization of a Gaussian
- $Z = \int f(z)dz \approx f(z_0)\int \exp\left\{-\frac{A}{2}(z - z_0)^2\right\}dz =$

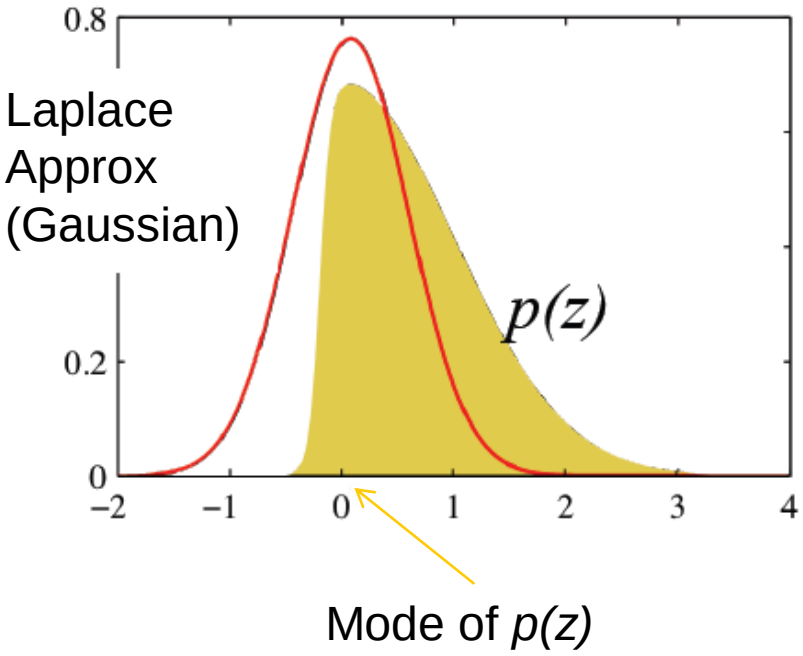
  $f(z_0)\frac{(2\pi)^{1/2}}{A^{1/2}}$
- $q(z) = \frac{1}{Z}f(z_0)\exp\left\{-\frac{A}{2}(z - z_0)^2\right\}$

$$= \left(\frac{A}{2\pi}\right)^{1/2} \exp\left\{-\frac{A}{2}(z - z_0)^2\right\}$$

- ➔ Univariate normal
  with mean = z_0 and variance = $\frac{1}{A}$

Laplace Approximation Example

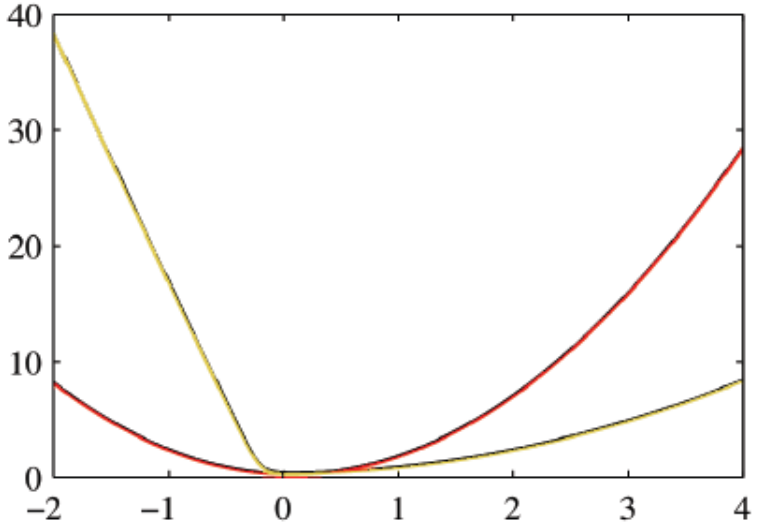
Applied to distribution $p(z) \propto \exp(-\frac{z^2}{2})\sigma(20z+4)$, where σ is sigmoid



Laplace Approx (Gaussian)

Mode of *p(z)*

Negative Logarithms



Gaussian approximation will only be well-defined if its precision A>0, or second derivative of *f(z)* at point z_0 is negative

Laplace Approximation: *M*-dimensions

- Task: approximate $p(z) = f(z)/Z$ defined over M-dimensional space of z
- At stationary point z_0 the gradient $\nabla f(z)$ vanishes
- Expanding around this point

$$\ln f(z) \cong \ln f(z_0) - \frac{1}{2}(z - z_0)^T A (z - z_0)$$

  - where A is the MxM Hessian matrix $A = -\nabla\nabla \ln f(z)\,|_{z=z_0}$
- Taking exponentials

$$f(z) \cong f(z_0) \exp\left\{-\frac{1}{2}(z - z_0)^T A (z - z_0)\right\}$$

Normalized Multivariate Laplacian

- $Z = \int f(z)dz \approx f(z_0) \int \left\{ -\frac{1}{2}(z - z_0)^T A(z - z_0) \right\} dz$

$$= f(z_0) \frac{(2\pi)^{M/2}}{|A|^{1/2}}$$

- Distribution *q(z)* is proportional to *f(z)* as
- $q(z) = \frac{1}{Z} f(z) = \frac{|A|^{1/2}}{(2\pi)^{M/2}} \exp\left\{ -\frac{1}{2}(z - z_0)^T A(z - z_0) \right\}$

$$= N(z|z_0, A^{-1})$$

Steps in Applying Laplace Approx.

- Find mode of z_0
  - Run a numerical optimization algorithm
- Evaluate Hessian matrix A at that mode.
- Multimodal distributions lead to different Laplace approximations depending on mode considered.

Weakness of Laplace Approx.

- Directly applicable only to real variables
  - Based on Gaussian distribution
- May be applicable to transformed variable
  - If $0 \leq \tau < \infty$ then consider Laplace approx of $\ln \tau$
- Based purely on a specific value of the variable
- Variational methods have a more global perspective

Model Evidence

- Recall that we are maximizing likelihood (or minimizing negative log-likelihood) to learn parameters for logistic regression.
- Likelihood function for logistic regression:
- $p(t|w) = \prod_{n=1}^{N} y_n^{t_n}\{1 - y_n\}^{1-t_n}$
- $y_n = \sigma(w^T \phi_n)$
- Maximizing likelihood leads to overfitting.
- Consider the example of Taipei 101 (台北101) and Toad Mountain (蟾蜍山)

Taipei 101 vs. Toad Mountain

- M_1: 台北101 (509M)
- D: Data
- θ_1: parameter vector
- Likelihood: $p(D|\theta_1, M_1)$
- Evidence: $p(D|M_1) = \int_{\theta_1} P(D|\theta_1, M_1)p(\theta_1|M_1)d\theta_1$

- M_2: 蟾蜍山 (128M)
- D: Data
- θ_2: parameter vector
- Likelihood: $p(D|\theta_2, M_2)$
- Evidence: $p(D|M_2) = \int_{\theta_2} P(D|\theta_2, M_2)p(\theta_2|M_2)d\theta_2$

Model Evidence (Cont'd.)

- Compared to $p(D|\theta_i, M_i)$, $p(D|M_i)$ is usually considered as a better measure on how good the model fit the data.

- Why?
- If we are doing MLE, we are searching for a single $\theta_{i_{MLE}}$ that maximize$p(D|\theta_i, M_i)$. This method is prone to overfitting because if we introduce a lot of features, then it is much easier to find a good θ_i that gives good $p(D|\theta_i, M_i)$.
- We thus need to penalize large search space, and only allow larger search space if there are enough data.
- This could be done naturally by looking at $p(D|M_i) = \int_{\theta_i} P(D|\theta_i, M_i)p(\theta_i|M_i)d\theta_i$ instead of $p(D|\theta_i, M_i)$ along.

Selecting Models Using Model Evidence

- Model evidence: $p(D|M_i) = \int_{\theta_i} P(D|\theta_i, M_i)p(\theta_i|M_i)d\theta_i$
- Likelihood: $p(D|\theta_i, M_i)$
- Note that model evidence is a function of model setting. It is not a function of parameters!
- For a set of models M_1, M_2, …, M_g to select from, compute $P(D|M_i)$ for each model, and select the one with the largest model evidence.
- Typical scenario:
  - Training logistic regression using 100, 200, 500, 1000 features. Want to determine which model setting is the best.
  - Selecting the best hyper-parameters for a model (e.g., choosing regularization parameters).

Hyper-parameter Tuning

- So far we have discussed two methods for hyper-parameter tuning.
- Method 1: Subtraining, tuning, test split.
- Method 2: Evidence function.

- What are the differences?
- Method 1 needs a three-way data split + Grid search.
- Method 2 only needs a training and test split.
  - No need to further split training data into subtraining and tuning. Hyper-parameter tuning can be done by looking at evidence function values of training data directly.
  - For a particular type of model, we can derive approximate functions for evidence functions so that it becomes easier to use.

Approximating Evidence Function

- Recall: Model evidence: $p(D|M_i) = \int_{\theta_i} P(D|\theta_i, M_i)p(\theta_i|M_i)d\theta_i$
- Omitting M_i, $p(D) = \int p(D|\theta)p(\theta)d\theta$
- The goal is to compute $p(D)$, which is often difficult to do.
- We can apply Laplace approximation to $p(D|\theta)p(\theta)$ so that the integral can be done analytically.
- First find the mode of $p(D|\theta)p(\theta)$, θ_{map}.
- Apply second order Taylor approximation.

Approximating Evidence Function (Cont'd.)

- $p(D) = \int p(D|\theta)p(\theta)d\theta$
- $\approx p(D|\theta_{map})p(\theta_{map})$

$$\int \exp\left\{-\frac{1}{2}(\theta - \theta_{map})^T A(\theta - \theta_{map})\right\} d\theta$$

- $= p(D|\theta_{map})p(\theta_{map})\frac{(2\pi)^{M/2}}{|A|^{1/2}}$
- Taking natural logarithm:
- $\ln p(D)$
- $\approx \ln p(D|\theta_{map}) + \ln p(\theta_{map}) + \frac{M}{2}\ln 2\pi - \frac{1}{2}\ln|A|$

Approximating Evidence Function (Cont'd.)

- Taking natural logarithm on the approximated function:
- $\ln p(D)$
- $\approx \ln p(D|\,\theta_{map}) + \underbrace{\ln p(\theta_{map}) + \frac{M}{2}\ln 2\pi - \frac{1}{2}\ln|A|}$

Occam factor that penalizes model complexity

- where θ_{map} is the value of θ at the mode of the posterior
- A is Hessian of second derivatives of negative log posterior

Approximating Evidence Function (Cont'd.)

- Consider the Hessian matrix

$$H = -\frac{\ln p(D|\theta)}{\partial\theta\partial\theta^T} = -\sum_{n=1}^{N}\frac{\ln p(d_i|\theta)}{\partial\theta\partial\theta^T} = \sum_{n=1}^{N} H_i$$

  where $D = \{\mathrm{d}_1, \mathrm{d}_2, \ldots, \mathrm{d}_\mathrm{n}\}$ is the collection of N data points; θ is a M by 1 matrix.

- Let $\widehat{H} = \frac{1}{N}\sum H_i$
- $\ln|H| = \ln|N\widehat{H}| = \ln N^M\left|\widehat{H}\right| = M\ln N + \ln\left|\widehat{H}\right|$
- As N increases, the second term will not grow ➔ drop it

$$\ln|H| \approx M\ln N$$

Bayes Information Criterion (BIC)

Assuming broad Gaussian prior over parameters & Hessian is of full rank,
approximate model evidence is

$$\ln p(D) = \ln p(D|\theta_{map}) + \ln p(\theta_{map}) + \frac{M}{2}\ln 2\pi - \frac{1}{2}\ln|H|$$
$$\approx \ln p(D|\theta_{map}) - \frac{1}{2}M \ln N$$

- N is the number of data points
- M is the number of parameters in θ
- Compared to Akaike Information Criteria (AIC) given by $\ln p(D|w_{ML}) - M$

  BIC penalizes model complexity more heavily

How to use BIC (and AIC) (Con't.d)

- Given a training data set.
- Note: you do not need extra tuning or test data set to apply BIC.
- Note: If you need prediction performance, then a test dataset is required. However, if you just want to train and pick the best model, then on test data is not required.

- Consider the following example.
- Three model settings: M_1: 100 features; M_2: 200 features; M_3: 500 features.

How to use BIC (and AIC)

- Train M_1 using likelihood maximization (or Error minimization), the trained model is $\theta_{map,1}$
- $BIC_1 = \ln p(D|\theta_{map,1}) - \frac{1}{2} M \ln N$
  - M: number of feature
  - N: number of data points
- Repeat the process for three models, compare BIC_1, BIC_2, BIC_3.
- Pick the model setting with the largest BIC.

Weakness of AIC, BIC

- AIC and BIC are easy to evaluate
- But can give misleading results since
  - Hessian matrix may not have full rank since many parameters not well-determined
- Can obtain more accurate estimate from

$$\ln p(D) = \ln p(D|\theta_{map}) + \ln p(\theta_{map}) + \frac{M}{2}\ln 2\pi - \frac{1}{2}\ln|A|$$

Summary

- Laplace approximation fits the best Gaussian to the data.
- Defined for both univariate and multivariate.
- Laplace approximation can be used to derive BIC criterion.
- AIC and BIC are simple but may be inaccurate.

BAYESIAN LOGISTIC REGRESSION

Topics

- Roadmap of Bayesian Logistic Regression
- Laplace Approximation
- Evaluation of posterior distribution
  - Gaussian approximation
- Predictive Distribution
  - Convolution of Sigmoid and Gaussian
  - Approximate sigmoid with probit

Recap of Logistic Regression

- Feature vector ϕ , two-classes C_1 and C_2
- *A posteriori* probability $p(C_1|\phi)$ can be written as $p(C_1|\phi) = y(\phi) = \sigma(w^T\phi)$

  where ϕ is a *M*-dimensional feature vector

  $\sigma(\cdot)$ is the logistic sigmoid function
- Goal is to determine the *M* parameters
- Potential problem:
  - Overfitting with large number of features.
  - Overfitting for nearly separable datasets.

Roadmap of Bayesian Logistic Regression

- Logistic regression is discriminative probabilistic linear classification $p(C_1|x) = \sigma(w^T\phi)$
- Exact Bayesian inference for Logistic Regression is intractable

1. Evaluation of posterior distribution $p(w|t)$
   - Needs normalization of prior $p(w) = N(w|m_0, s_0)$ times likelihood (a product of sigmoids)
     $p(t|w) = \prod_{n=1}^{N} y_n^{t_n}\{1 - y_n\}^{1-t_n}$
   - Apply Laplace approximation to get Gaussian
2. Evaluation of predictive distribution ($p(w|t) \approx q(w)$)
   $p(C_1|\phi, t) \cong \int \sigma(w^T\phi)q(w)dw$
   - Approximate Sigmoid by Probit

Evaluation of Posterior Distribution

- Gaussian prior $p(w) = N(w|m_0, S_0)$
  - where m_0 and S_0 are hyper-parameters
- Posterior distribution $p(w|t) \propto p(w)p(t|w)$
  - where $t = (t_1, \ldots, t_N)^T$
- Substituting $p(t|w) = \prod_{n=1}^{N} y_n^{t_n}\{1 - y_n\}^{1-t_n}$
  - where $y_n = \sigma(w^T \phi_n)$
- $\ln p(w|t) = -\frac{1}{2}(w - m_0)^T S_0^{-1}(w - m_0)$
  $+ \sum_{i=1}^{n}[(t_n \ln y_n) + (1 - t_n)\ln(1 - y_n)] + const$
- Usually set $m_0 = 0, S_0 = \frac{1}{\lambda} I$

Finding Posterior Mode: $p(w|t)$

- $\ln p(w|t) = -\frac{\lambda}{2} w^T w + \sum_{i=1}^{n} [(t_n \ln y_n) + (1 - t_n) \ln(1 - y_n)] + const$
- To be consistent with our previous discussion, define $E(w) = -\ln p(w|t)$
- Apply $w^{\tau+1} = w^{\tau} - H^{-1} \nabla E_n$ (*Newton-Raphson Numerical Optimization*)
- $\nabla E(w) = \lambda w + \sum_{n=1}^{N} (y_n - t_n) \phi_n = \lambda w + \Phi^T (y - t)$
- Hessian: $\nabla \nabla E(w) = \lambda I + \sum_{n=1}^{N} y_n (1 - y_n) \phi_n \phi_n^T = \Phi^T R \Phi + \lambda I$
  - Φ is NxM design matrix whose n[th] row is ϕ_n^T
  - R is NxN diagonal matrix with element $R_{nn} = y_n (1 - y_n)$

Hessian is not constant and depends on w through R
Since H is positive-definite (i.e., for arbitrary u, $u^T Hu > 0$), error function is a concave function of w and so has a unique minimum

Newton-Raphson Update

- $w^{(new)} = w^{(old)} - H^{-1}\nabla E(w)$
- $\nabla E(w) = \lambda w^{(old)} + \Phi^T(y - t)$
- Hessian: $\nabla\nabla E\left(w^{(old)}\right) = \Phi^T R\Phi + \lambda I$
- ➔ $w^{(new)} = w^{(old)} - (\Phi^T R\Phi + \lambda I)^{-1}\left[\lambda w^{(old)} + \Phi^T(y - t)\right]$
- Iterate until converge ➔ w_0 (mode of $p(w|t)$)

Laplace Approximation (summary)

- Need mode w_0 of posterior distribution $p(w|t)$
  - Done by a numerical optimization algorithm
- Fit a Gaussian centered at the mode:

$$q(w) = \frac{1}{W} f(w) = \frac{|A|^{1/2}}{(2\pi)^{M/2}} \exp\left\{-\frac{1}{2}(w - w_0)^T A (w - w_0)\right\}$$

- Meaning: The posterior of $w \sim \mathrm{N}(w_0, A^{-1})$
- Needs the second derivatives of log posterior $p(w|t)$
- $$A = S_N^{-1} \equiv -\nabla\nabla \ln p(w|t) = S_0^{-1} + \sum_{i=1}^{n} y_n(1 - y_n)\phi_n \phi_n^T$$
$$= \lambda I + \Phi^T R \Phi$$

Gaussian Approximation of Posterior

- Maximize posterior *p(*w|t*)* to give
  - MAP solution w_{map} :Done by numerical optimization
  - Defines mean of the Gaussian
- Fit a Gaussian centered at the mode $q(w) = \frac{1}{W} f(w) = \frac{|A|^{1/2}}{(2\pi)^{M/2}} \exp\left\{-\frac{1}{2}\left(w - w_{map}\right)^T A\left(w - w_{map}\right)\right\}$
- Covariance given by
  - $A = S_N^{-1} \equiv -\nabla\nabla \ln p(w|t) = S_0^{-1} + \sum_{i=1}^{n} y_n(1 - y_n)\phi_n\phi_n^T = \lambda I + \Phi^T R \Phi$
- Gaussian approximation to posterior: $q(w) = N(w|w_{map}, S_N)$

Approximately Evidence Maximization

- Recall for a model: $p(D) = \int p(D|\theta)p(\theta)d\theta$
- $\ln Z = \ln p(D) =$

$$\ln p(D|\,\theta_{map}) + \ln p(\theta_{map}) + \frac{M}{2}\ln 2\pi - \frac{1}{2}\ln|A|$$

- In our case, for a given λ, we have
- Likelihood: $p(t|w) = \prod_{n=1}^{N} y_n^{t_n}\{1-y_n\}^{1-t_n}$
- Prior: $p(w) = N(w|0, \frac{1}{\lambda}I)$
- $\ln Z =$

$$\ln p\left(t\middle|w = w_{map}\right) + \ln N\left(w_{map}\middle|0, \frac{1}{\lambda}I\right) + \frac{M}{2}\ln 2\pi + \frac{1}{2}\ln|S_N|$$

- $S_N^{-1} = S_0^{-1} + \sum_{i=1}^{n} y_n(1-y_n)\phi_n\phi_n^T = \lambda I + \Phi^T R\Phi$

Approximately Evidence Maximization

- Fixing w_{map} and R, want to maximizing $\ln Z$ with respect to λ.
- Recall: $\ln Z =$
  $\ln p(t|w = w_{map}) + \ln N\left(w_{map}\middle|0, \frac{1}{\lambda} I\right) + \frac{M}{2}\ln 2\pi - \frac{1}{2}\ln|S_N^{-1}|$
  - $\ln N\left(w_{map}\middle|0, \frac{1}{\lambda} I\right) = \frac{M}{2}\ln\lambda - \frac{M}{2}\ln 2\pi - \frac{\lambda}{2} w_{map}^T w_{map}$
  - $S_N^{-1} = \lambda I + \Phi^T R \Phi$
- $\frac{\partial \ln Z}{\partial \lambda} = \frac{M}{2\lambda} - \frac{1}{2} w_{map}^T w_{map} - \frac{1}{2}\sum_{i=1}^{M} \frac{1}{\lambda_i + \lambda} = 0$
  - $\lambda_1, \lambda_2, \ldots, \lambda_M$ are the eigenvalues of $\Phi^T R \Phi$.
- ➔$\sum_{i=1}^{M} \frac{\lambda_i}{\lambda_i + \lambda} \equiv \gamma = \lambda w_{map}^T w_{map}$
- Start with an initial λ➔ Compute γ➔ New $\lambda = \frac{\gamma}{w_{map}^T w_{map}}$
- Iterate for a fixed number of iteration to get new λ, then find the new w_{map}, and repeat.

Predictive Distribution

- Predictive distribution for class C_1, given new feature vector $\phi(x)$
  - Obtained by marginalizing w.r.t. posterior $p(w|\boldsymbol{t})$
- $p(C_1|\phi, \boldsymbol{t}) = \int p(C_1|\phi, \boldsymbol{t}, w)p(w|\boldsymbol{t})dw$ (Product rule)
- $= \int p(C_1|\phi, w)p(w|\boldsymbol{t})dw$ Given ϕ and w, C_1 is indep. of $\boldsymbol{t}$
- $\cong \int \sigma(w^T\phi)q(w)dw$ Approximate $p(w|t)$ by Gaussian $q(w)$

- Corresponding probability for class C_2:
- $p(C_2|\phi, t) = 1 - p(C_1|\phi, t)$

Predictive Distribution

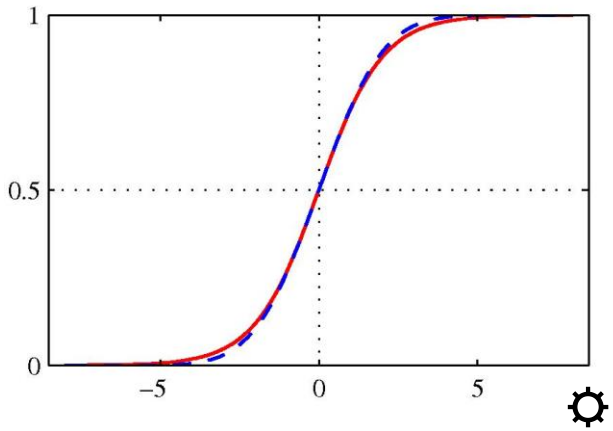
- Since $w \sim N(w_{map}, S_N)$, the variable $a = w^T\phi$ is also distributed normally with

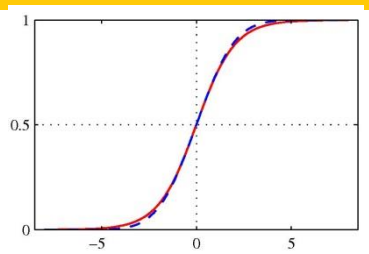
$$\mathrm{u_a} = E(a) = E(w^T\phi) = w_{map}^T\phi$$

$$\sigma_a^2 = Var(a) = Var(\phi^T w) = E(\phi^T(w - \overline{w})(w - \overline{w})^T\phi)$$

$$= \phi^T S_N \phi$$

- Predictive distribution is $p(C_1|t) = \int \sigma(a)p(a)da = \int \sigma(a)N(a|\mu_a, \sigma_a^2)da$
- Convolution of Sigmoid-Gaussian is intractable
- Use probit instead of logistic sigmoid





Approximation using Probit

- $p(C_1|t) = \int \sigma(a) N(a|\mu_a, \sigma_a^2) da$
- Use probit which is similar to Logistic sigmoid
  - Defined as $\Phi(a) = \int_{-\infty}^{a} N(\theta|0,1) d\theta$
- Approximate $\sigma(a)$ by $\Phi(\lambda a)$
  - Approximate $\sigma(a)$ by $\Phi(\lambda a)$
  - Find suitable value of λ by requiring that two have same slope at origin, which yields $\lambda^2 = \pi/8$
- Convolution of probit with Gaussian is a probit
  - $\int \Phi(\lambda a) N(a|\mu, \sigma^2) da = \Phi\left(\frac{\mu}{(\lambda^{-2}+\sigma^2)^{1/2}}\right) = \Phi\left(\frac{\mu}{\left(\frac{8}{\pi}+\sigma^2\right)^{\frac{1}{2}}}\right)$
- Thus, $p(C_1|t) = \int \sigma(a) N(a|\mu_a, \sigma_a^2) da \cong \Phi\left(\frac{\mu_a}{\left(\frac{8}{\pi}+\sigma_a^2\right)^{\frac{1}{2}}}\right)$

  $= \Phi\left(\mu_a \sqrt{\frac{\pi}{8}}\left[1 + \frac{\pi\sigma_a^2}{8}\right]^{-\frac{1}{2}}\right) \cong \sigma(k(\sigma_a^2)\mu_a)$
  - $k(\sigma^2) = (1 + \pi\sigma^2/8)^{-1/2}$

Probit Classification

- Applying it to $p(C_1|t) = \int \sigma(a) N(a|\mu_a, \sigma_a^2) da$
- We have $p(C_1|\phi, t) = \sigma(k(\sigma_a^2)\mu_a)$,
  - With $k(\sigma^2) = (1 + \pi\sigma^2/8)^{-1/2}$
  - $\mu_a = w_{map}^T \phi$, $\sigma_a^2 = \phi^T S_N \phi$
- Decision boundary corresponding to $p(C_1|\phi, t) = 0.5$ is given by $\mu_a = 0$
- This is the same solution as $w_{map}^T \phi = 0$
- Thus marginalization has no effect!
- When minimizing misclassification rate with equal prior probabilities
- For more complex decision criteria it plays important role

Summary

- Logistic regression is a linear probabilistic discriminative model $p(C_1|x) = \sigma(w^T\phi)$
- Bayesian Logistic Regression is intractable
- Using Laplacian the posterior parameter distribution $p(w|\mathrm{t})$ can be approximated as a Gaussian
- Predictive distribution is convolution of sigmoids and Gaussian
  - Probit yields convolution as probit: $p(C_1|\phi, t) \cong \int \sigma(w^T\phi)q(w)dw$